Testing Software for Fairness
October 2, 2018

CS 520
Theory and Practice of Software Engineering
Fall 2018

Coming up

• Guest lecture tomorrow (Wednesday), October 3
  4-5PM in CS 151
  – Please attend.
  (if you have a time conflict – we’ll post a video)
• No class on Thursday, October 4
• Extra credit assignment posted.

Assignments

• Homework 1 (due Oct 16):
  https://people.cs.umass.edu/~brun/class/2018Fall/CS520/hw1.pdf
• Final project assignment:
  https://people.cs.umass.edu/~brun/class/2018Fall/CS520/finalProject.pdf
• End of class today, time to discuss groups

CS 520
Extra Credit
Debugging Study

This semester, we will offer an optional extra credit assignment involving a study on debugging Java programs.

The extra-credit assignment will be due in person, during a two-hour one-on-one session. You will be presented with Java code written by someone else and asked to perform coding tasks using modern development tools.

How to sign up:

If you wish to participate in this extra credit assignment, you have to email bjohnson@cs.umass.edu with the subject:
CS 520 Extra Credit scheduling request
with the body of the email containing your name and a short message saying you would like to participate in the extra credit assignment. You will receive an email back with instructions on how to schedule your two-hour session.

This extra-credit assignment will be done on a first-come-first-served basis. While we anticipate that we will be able to accommodate everyone who wants to participate, if too many people wish to participate, we may stop administering the assignment after a certain point. Once you have scheduled a session, you are guaranteed to get a chance to participate.

Point value:

This assignment will be worth up to 2 points on your final grade. For reference, each of the in-class exercises is worth 7.5 points, so completing this extra credit is like a 26.7% boost to one in-class exercise's grade.

INSIDE THE GLISTENING RED CAVE

of the patient’s abdomen, surgeon Michael Stifelman carefully guides two robotic arms to tie knots in a piece of thread. He manipulates a third arm to drive a suturing needle through the fleshy mass of the patient’s kidney, stitching together the hole where a tumor used to be. The final arm holds the endoscope that streams visuals to Stifelman’s display screens. Each arm enters the body through a tiny incision about 5 millimeters wide.

To watch this tricky procedure is to marvel at what can be achieved when robot and human work in tandem. Stifelman, who has done several thousand robot-assisted surgeries as director of NYU Langone’s Robotic Surgery Center, controls the robotic arms from a console. If he swivels his wrist and pinches his fingers closed, the instruments inside the patient’s body perform the same exact motions on a much smaller scale. “The robot is one with me,” Stifelman says as his mechanized appendages pull tight another knot.

PRECISE AND DEXTEROUS SURGICAL ROBOTS MAY OUTDO HUMAN SURGEONS

By ELIZA STRICKLAND

Trusting Robots

32 | jun 2016 | north America | SPectrum. ieee. orG

ILLUSTRATION BY Carl De Torres

Modern software influences critical decisions.

Software can make bad decisions.
Software can discriminate!
Rachael Tatman, “Gender and Dialect Bias in YouTube’s Automatic Captions” in 2017 Workshop on Ethics in Natural Language Processing.

You Tube Automatic captions

Rachael Tatman, “Gender and Dialect Bias in YouTube’s Automatic Captions” in 2017 Workshop on Ethics in Natural Language Processing.

Joy Buolamwini
https://www.ted.com/talks/joy_buolamwini_how_i_m_fighting_bias_in_algorithms

how people want to use vision software

today's goals

Define software discrimination.

Operationalize measuring discrimination through causal software testing.
Design software to be fair

Typically machine learning systems:
- Balance training sets
- Introduce training noise
- Constrain regression's loss function
- Split criteria on sensitive inputs

This talk is not about policy.

Fairness: prior definitions

1. Hide the data

- Ineffective because of data correlation.
  - Latanya Sweeney. Discrimination in online ad delivery. CACM 2013

2. Compare subpopulation proportions

- Ineffective if race or age correlate with savings or income
- Fails to identify discrimination against individuals

3. Correlation or mutual information

- Correlation does not measure causation

How group discrimination can fail

Europe
- Approve loans to all green applicants
- Approve loans to all purple applicants
Europe and Asian discriminations cancel each other out, and the group discrimination measure can be 0.

Asia
- Deny loans to all green applicants
- Deny loans to all purple applicants

Fairness: prior definitions

Correlation
- Corr(race, approved) = 0.8
- MI(race, approved) = 0.6

Correlation does not measure causation

What is fairness?

Sensitive inputs should not affect software behavior.

We want to measure causality!


causal testing

Sensitive inputs should not affect software behavior.

causal testing

No need for an oracle!

causal testing

Themis automated test-suite generator

How much does my software discriminate with respect to …?

Does my software discriminate more than 10% of the time, and against what?

Themis generates a test suite or can use a manually written one

http://fairness.cs.umass.edu

causal testing

discrimination measures

causal discrimination

LOAN( ) \not= \text{LOAN ( )}

group discrimination

apparent discrimination (causal or group)
Apparent discrimination can be group or causal, measured on a given test suite or operational profile.

Software may discriminate, but not for a given set of customers.

Fair software may appear to discriminate (e.g., Amazon same-day delivery).

Apparent discrimination can be group or causal, measured on a given test suite or operational profile.

Evaluation

Eight open-source decision systems trained on two public data sets:

| Discrimination-aware Logistic Regression | 88 |
| Discrimination-aware Decision Tree | 40 |
| Discrimination-aware Naive Bayes | 18 |
| Discrimination-aware Decision Tree | 91 |
| Naive Bayes | solid-learn |
| Decision Tree | |
| Logistic Regression | |
| SVM | |

Findings

Group discrimination is not enough.

- More than 11% of the individuals had the output flipped just by altering the individual's gender.

Decision tree trained not to group discriminate against gender causal discriminated against gender: 0.11.

Findings

Trying to avoid group discrimination may introduce other discrimination.

- Training a decision tree not to discriminate against gender made it discriminate against race 38.4% of the time.

Findings

Pruning is highly effective.

- The more a system discriminates, the more efficient Themis is.
- On average, pruning reduced test suites by 148x for causal and 2,849x for group discrimination. Best improvement was 13,000x.
related work

Ways of measuring discrimination
- CV score [19]
- correlation, mutual information [79]
- Output probability distributions [51]

Discrimination-aware algorithms [18, 40, 88, 91]

Measuring discrimination with manually-written tests [79]

Causal model inference [Maier et al., UAI’13]

Fairness verification [Albarghouthi et al., OOPSLA’17]

what’s next?

- Software with complex inputs, such as natural language or photographs and videos.
- What definition is right for what software requirements context?
- Efficiency in testing.

Contributions

http://fairness.cs.umass.edu

- Causality-based definition and method for measuring software fairness
- Themis, an automated test-suite generator for fairness testing
- Provably-sound pruning test-suite reductions
- Evaluation on real-world software, demonstrating Themis’ effectiveness

CS 520

Final project description

Final projects will be completed in teams of 1 or 2 members. Each team is responsible for a single project. You should choose a team and a project by Tuesday, October 5, 2010. Enhancements of selected projects will be accepted until the final project is due.

The final project is due on Tuesday, December 14, 2010. It should:

1. MSRI 2010 Mining Challenge
2. Replication study
3. Model Fairness for Learning Processes
4. Efficient, Decentralized Navigation
5. Self-defined software engineering research project

MSRI 2010 Mining Challenge

The MSRI 2010 Mining Challenge is an annual challenge in which they provide a dataset and ask you to answer research questions about the dataset. Read the description of this year’s dataset, research questions, and challenge here.

http://fairness.cs.umass.edu/2010-mine-challenge

Replication Study

A replication study involves an existing research paper, replicating its experiments on the same data, and then extending the experiments to expand on the questions the experiments answer. For this project, we highly recommend selecting a paper with publicly available dataset and code to reproduce the experiments. The project consists of a write-up describing the process of reproducing the experiments, deviations in the achieved results from the original ones reported in the paper, and lessons learned from applying the experiments to your dataset.

Here is a list of several papers that are good candidates for replication:

1. Automatic generation of test cases for exceptional behaviors from Verifiable Contracts
   Paper: https://pdfs.semanticscholar.org/8d55/40f0697d365af35b245e95bc467707e960f0.pdf
   Source code: testc Guardian
   Dataset: https://github.com/Guardian

2. Fairness-Aware Program Synthesis
   Paper: https://pdfs.semanticscholar.org/8d55/40f0697d365af35b245e95bc467707e960f0.pdf
   Source code: https://github.com/Alexey/Rykov
   Dataset: https://github.com/Alexey/Rykov